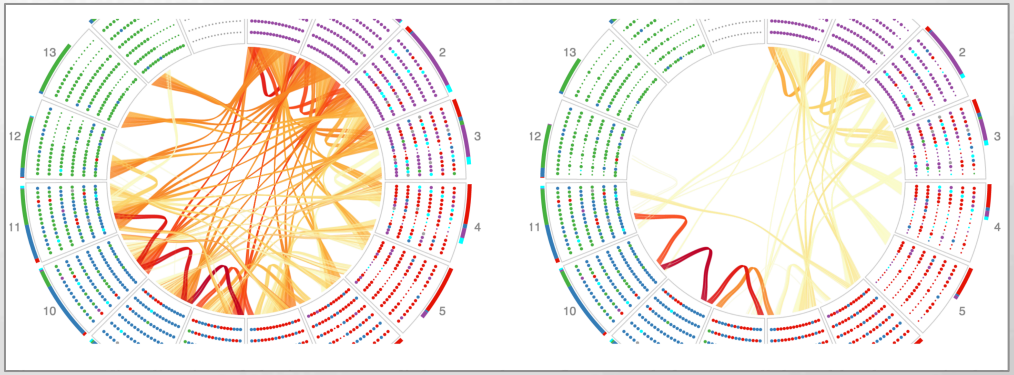


Towards Interactive Analysis and Exploration of the HPC Performance Landscape

Yarden Livnat[1], Valerio Pascucci[1], Timo Bremer[1,2], Abhinav Bhatele[2],Martin Schulz[2], Todd Gamblin[2], Kate Isaccs[2,3]

[1]CEDMAV, SCI Institute, University of Utah
[2]Center for Applied Computing, LLNL
[3]UC Davis

SCI INSTITUTE CEDMAV THE UNIVERSITY OF UTAH UCDAVIS UNIVERSITY OF CALIFORNIA ASC PIPER project

ON

DAMSELS AND DRAGONS

Exascale computing

…the light at the end of the tunnel
…the light of an incoming train

Challenges

Perform in resource constrained environment

Survive higher failure rates

Complex heterogeneous architectures

Efficiency

Increase science productivity vs. cost

Invariant

Exploiting the machines full capabilities is exponentially more difficult with each new generation of hardware

Resilience

When does resilience becomes a reliability problem?

When do users notice faults and errors ?

- software crash
- wrong solution
- smoke? (the prisoner 1967)
- what about erratic behavior?

At 30,000ft, are these resilience issues?

- Job interference
- User trust

Performance Analysis

Concerns with ***what***, ***where*** and ***when***

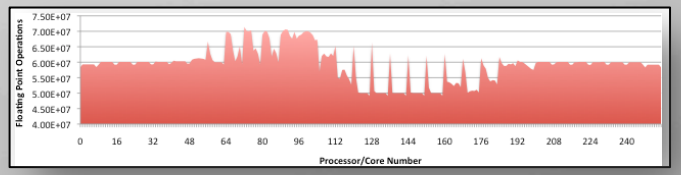
What are the important factors?

Where do they impact the most?

When do they impact?

We collect a lot of data

- Fine grain, low level data
- Hard to match with application structure
- Hard to comprehend



and yet the exascale data train is coming…

Challenges

Provide **comprehensive** data acquisition,

capture **holistic view** of the system,

Scalable

Yet not collect too much data:

- Hard to process
- More precise measurement effect the computation

Challenges

Cater to the end users

make performance analysis more **accessible**, more **intuitive**

make sense of massive amounts of disparate, incomplete and dynamic data

Interactive Exploratory Visualization

user-centric, focus on the human in the loop

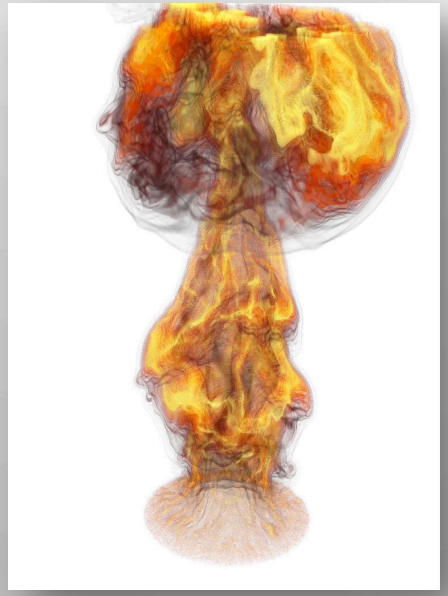
clear, concise visualizations

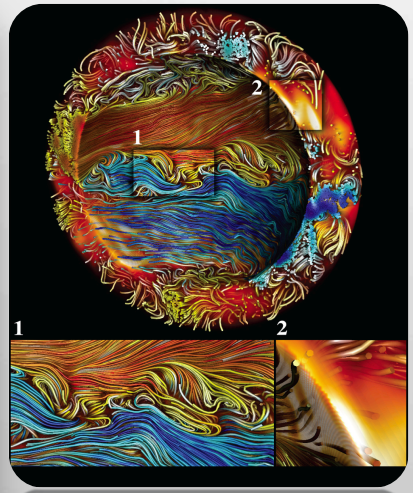
explore, comprehend, facilitate decision making

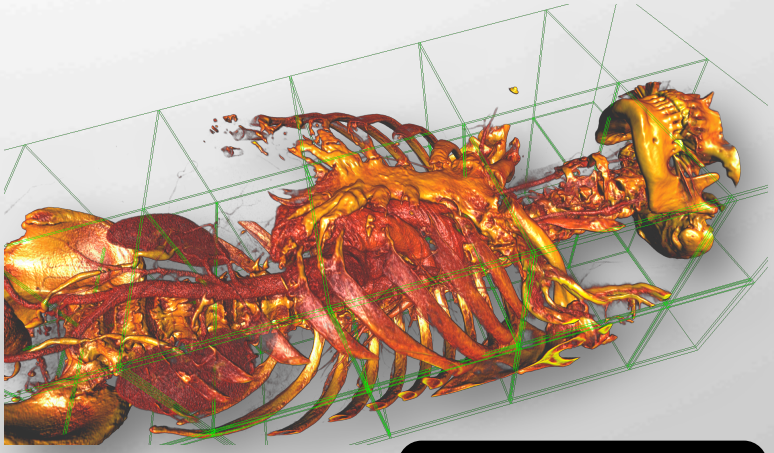
A few words on

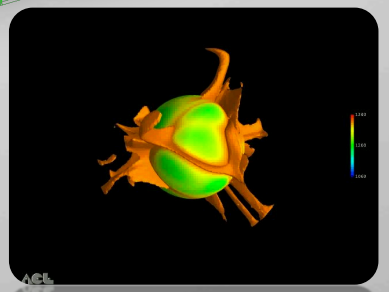
VISUALIZATION

Scientific visualization

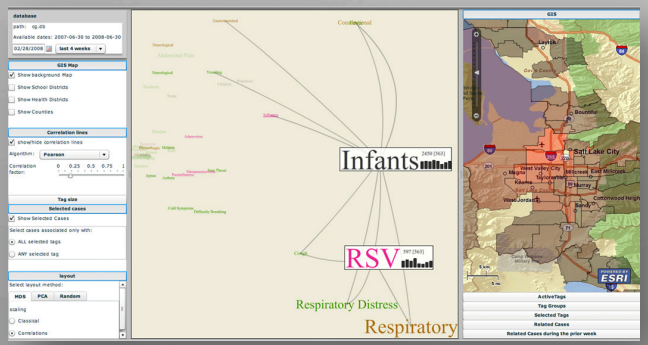


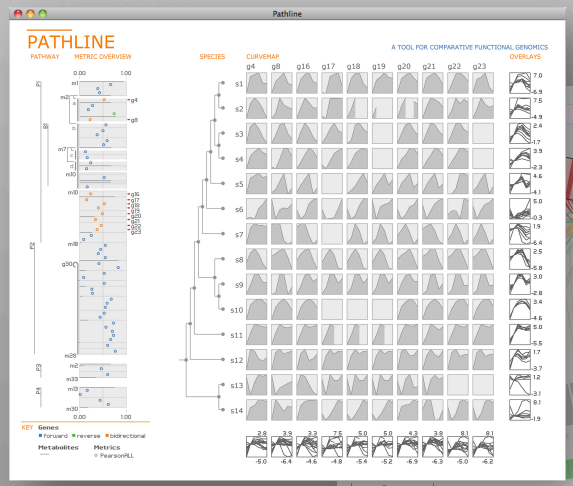


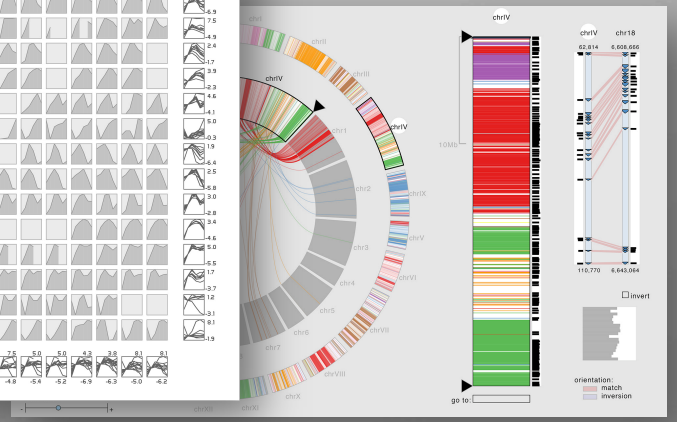




Information visualization



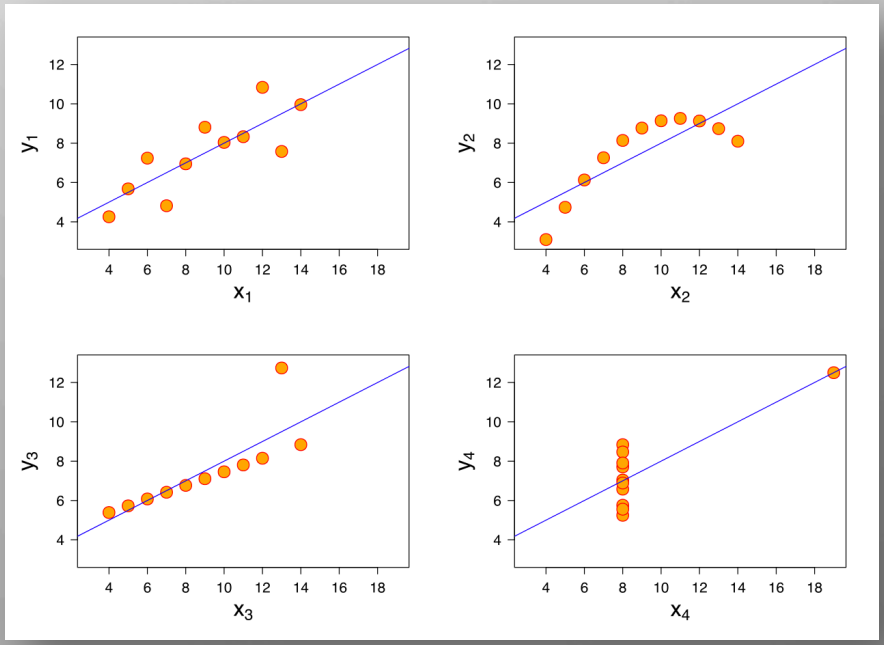




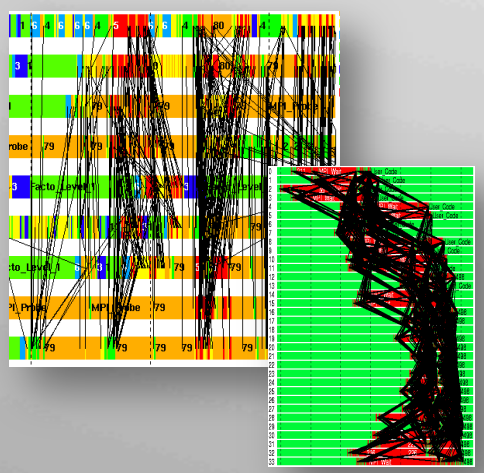
Anscombe's Quartet (Francis Anscombe, 1973)

Four datasets each with 11 points
Same statistical properties:

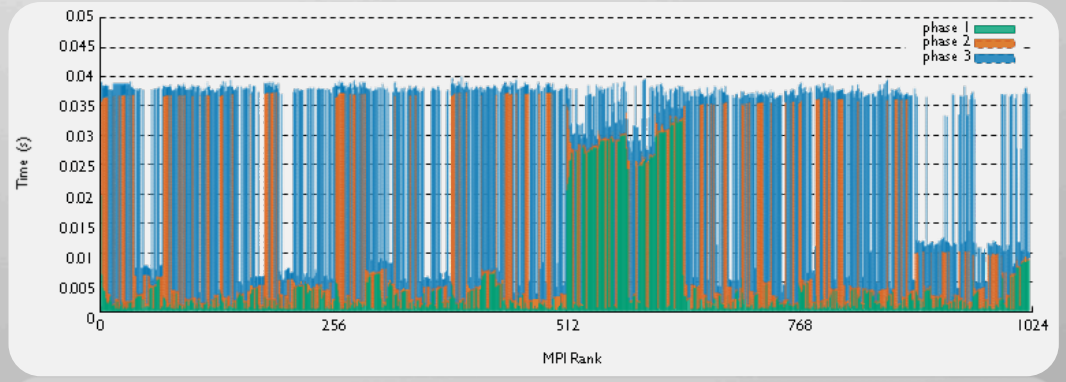
- Mean of x and y
- Variance of x and y
- Correlation between x and y
- Linear regression



We are used to static and aggregated visualizations



MPI Trace Data from runs with 16 and 34 processes

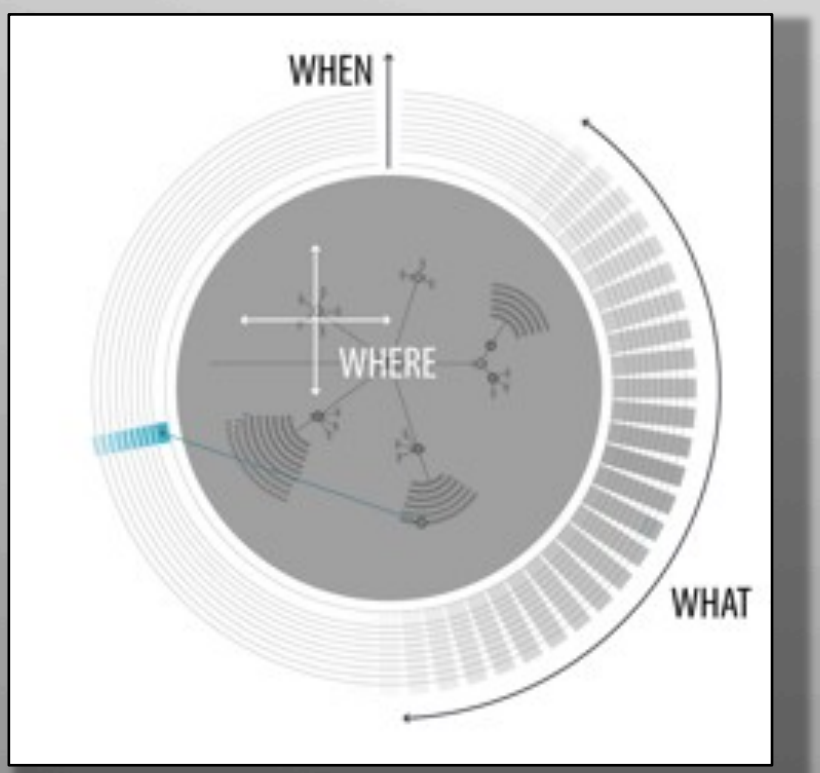


Times spent in the three load balancing sub-phases of a SAMRAI simulation plotted against the MPI ranks.

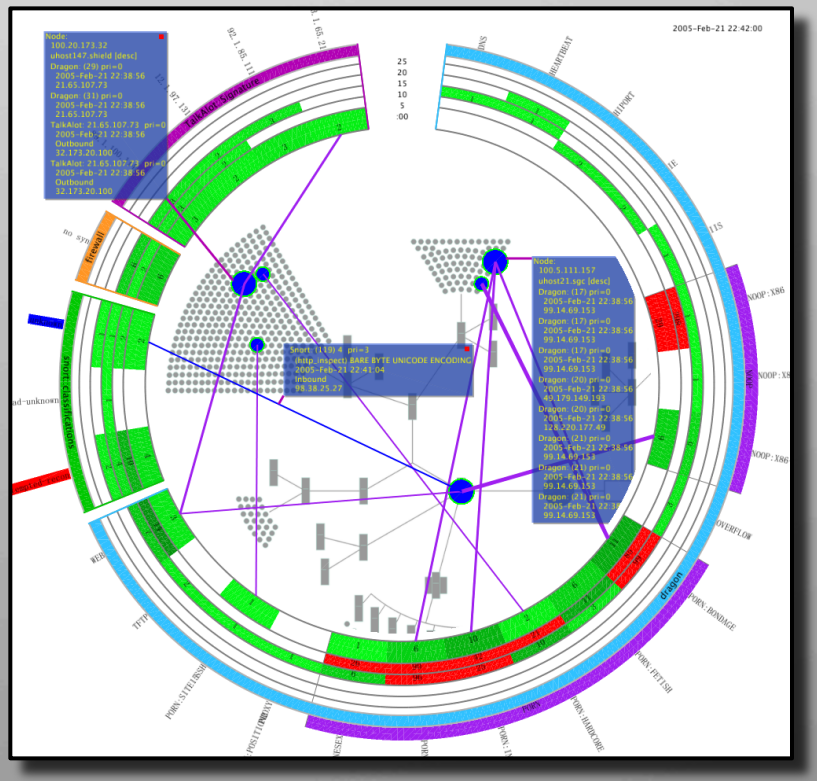
We can do better…

A quick example of the *what*, *where* and *when*
in the network intrusion detection domain

Traditional



VisAlert



Of course some domains are easier than others

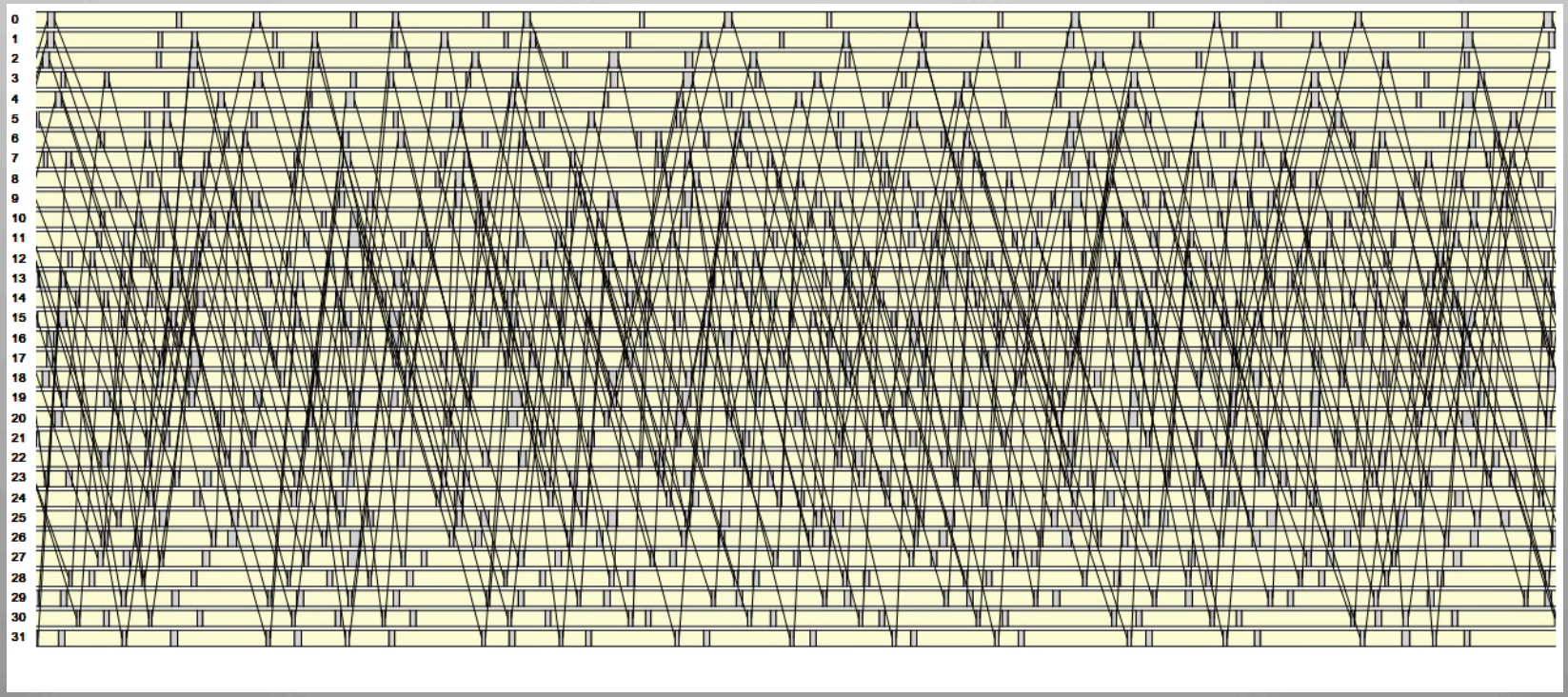
FROM INFORMATION TO INSIGHT

Ravel

Making Message Traces Readable

Trace visualization is a helpful tool to show message details
but **physical timeline** view can create a hairball

Ravel uses **logical timeline** to unravels the hairball

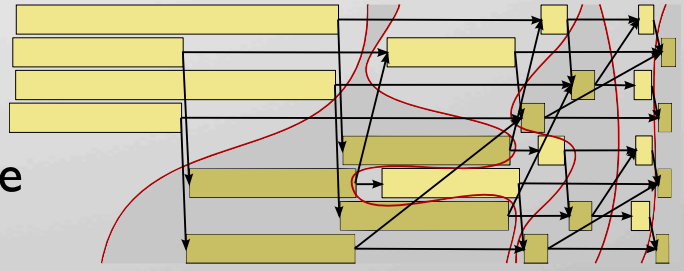


Ravel

Visualizing Traces in Logical Time

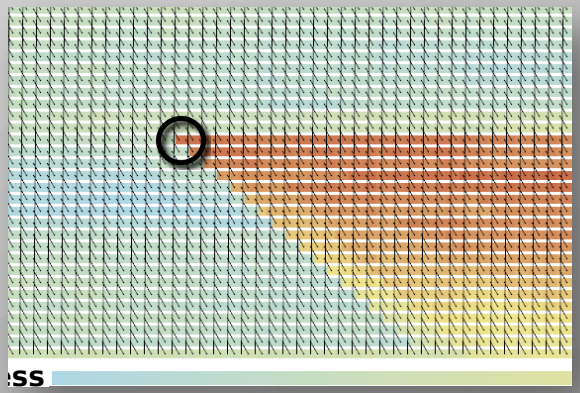
Identify time slices

- Based on connected components
- Start with send/recv pairs and grow from there
- Heuristics on when to stop growing



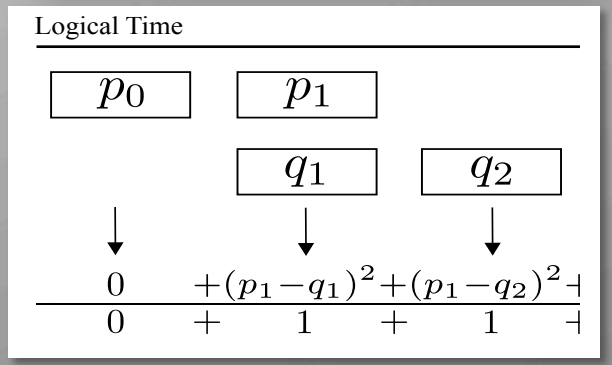
Map timing metrics

- Mapping to virtual time loses physical time
- Reintroduction of time using lateness metric
  - Time difference to end of aligned phase
  - Shows propagations of delays



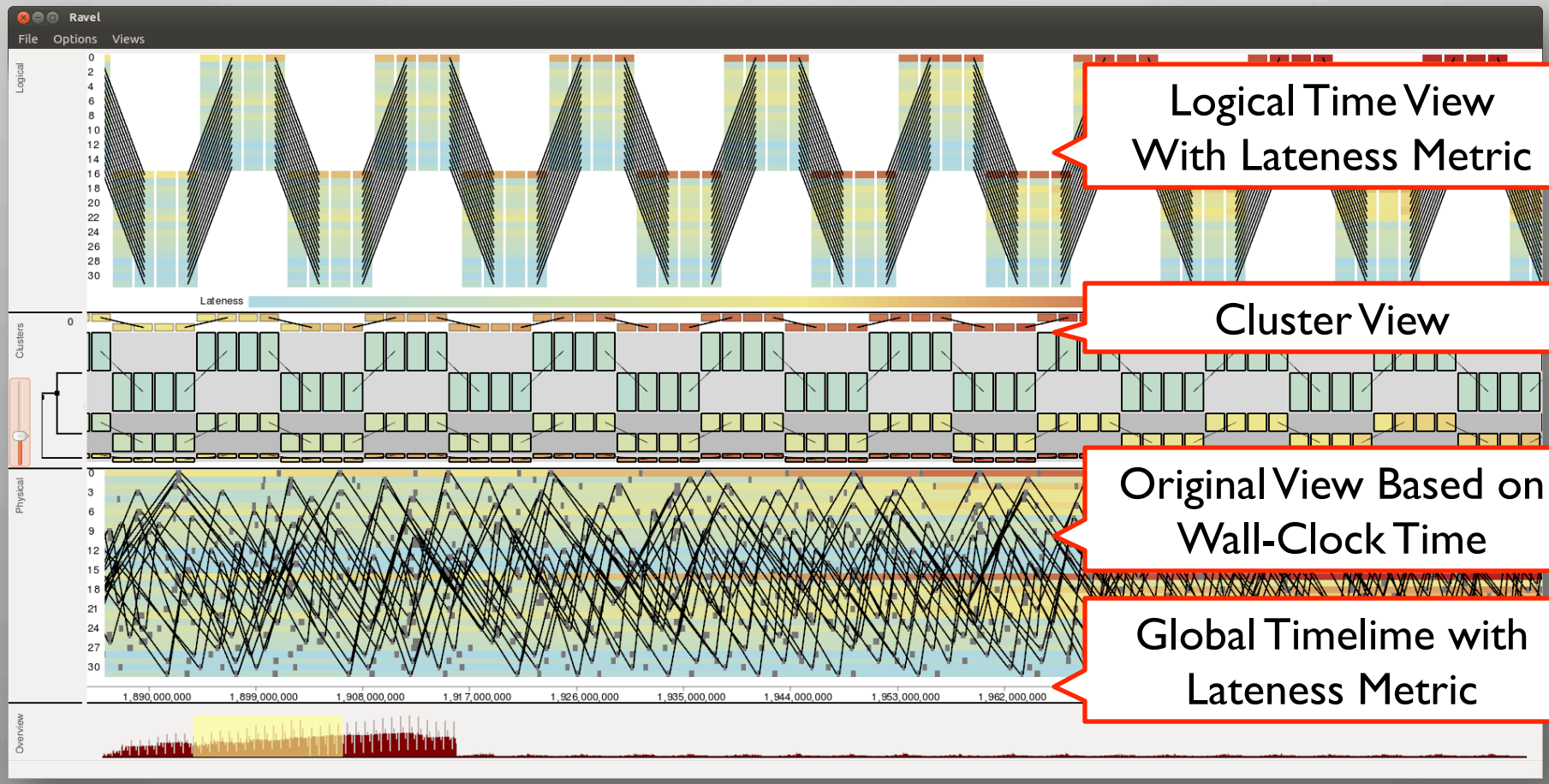
Cross process clustering

- Aggregate traces with similar lateness
- Use of representative traces to show data



Ravel

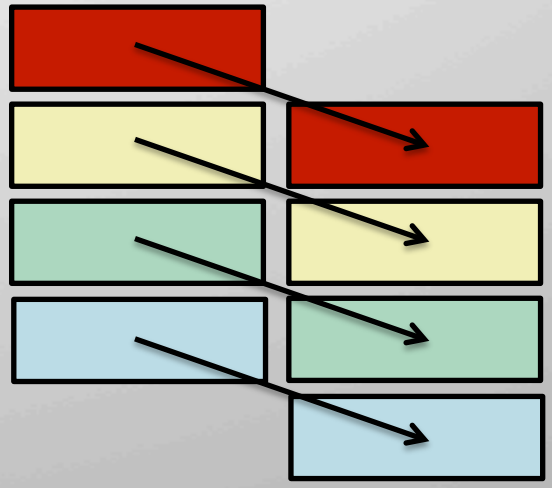
Visualizing Traces in Logical Time



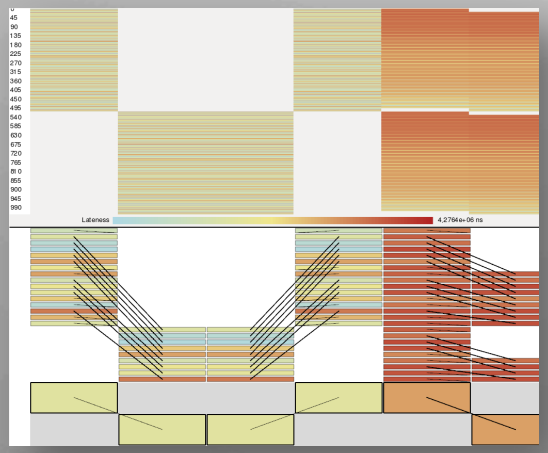
Ravel Case Study:
Optimizing Communication Patterns

Communication benchmark for physics simulation

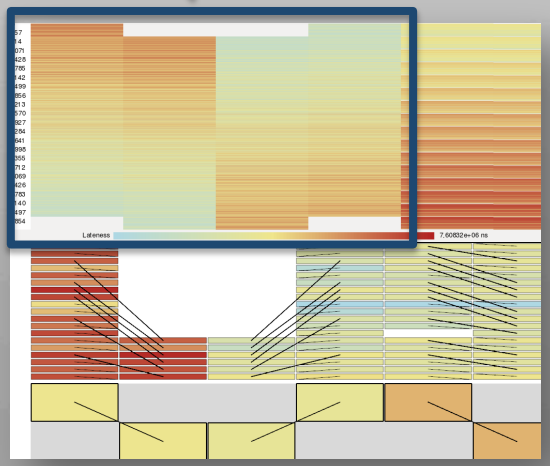
(blocking pf3d)



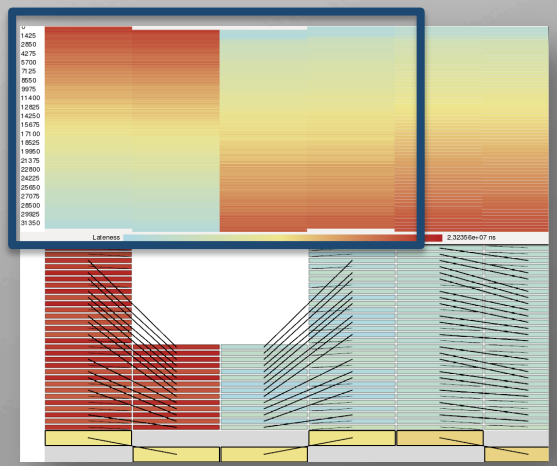
Inverted gradient of lateness



1k processes



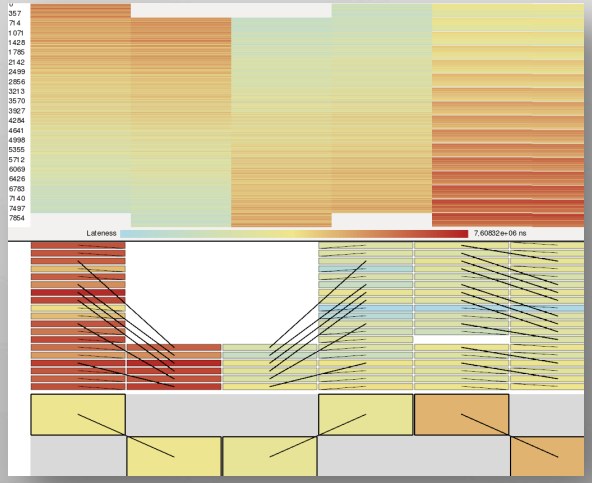
8k processes



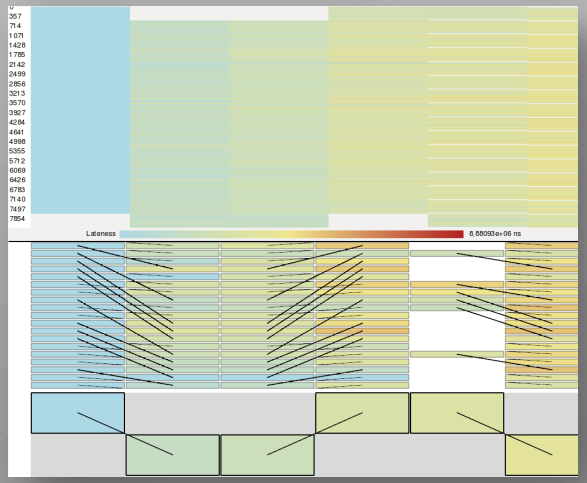
32k processes

Ravel Case Study:
Optimizing Communication Patterns

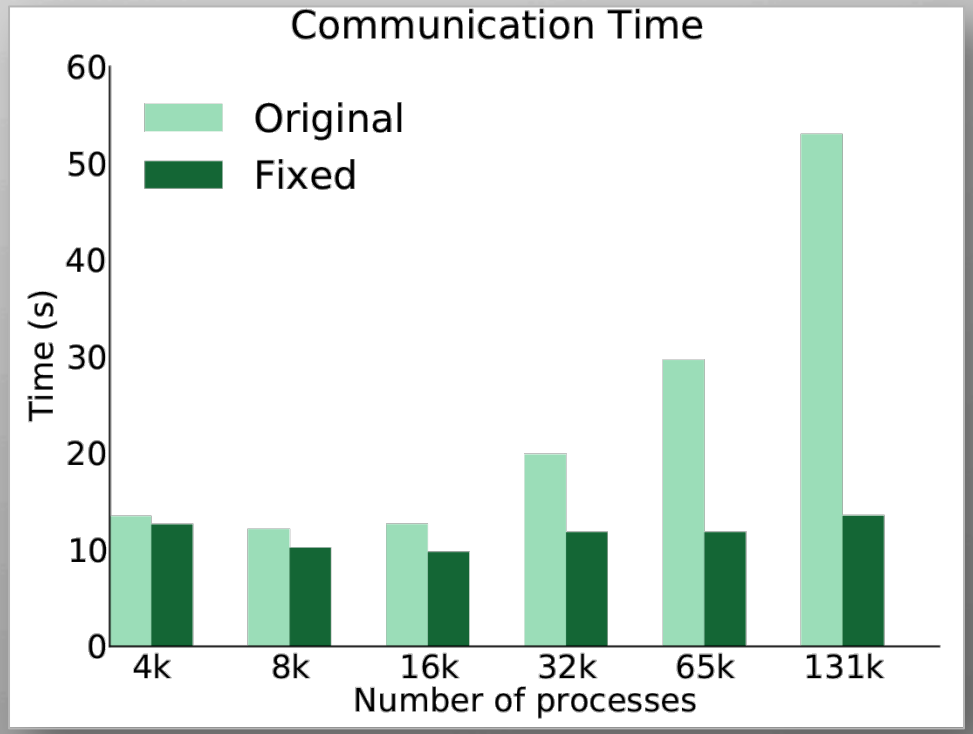
Before



After



Changed to an asynchronous communication
No waiting for a send before a receieve



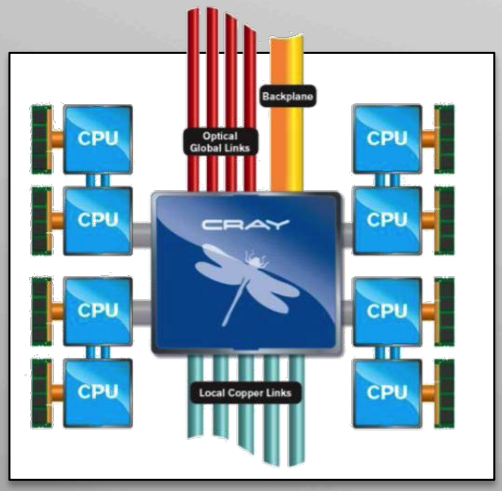
DAMSELS AND DRAGONS

Dragonfly

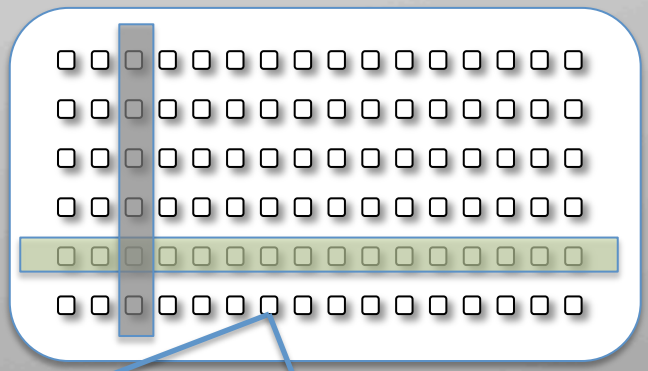
Edison: Cray XC30 at NERSC

130,000+ cores, 5500+ nodes, 1440 Aries routers, 2.57 Pflops

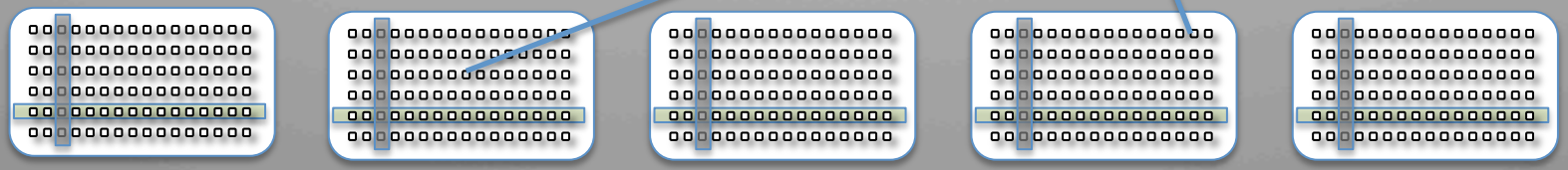
Aries router has adaptive non-minimal routing



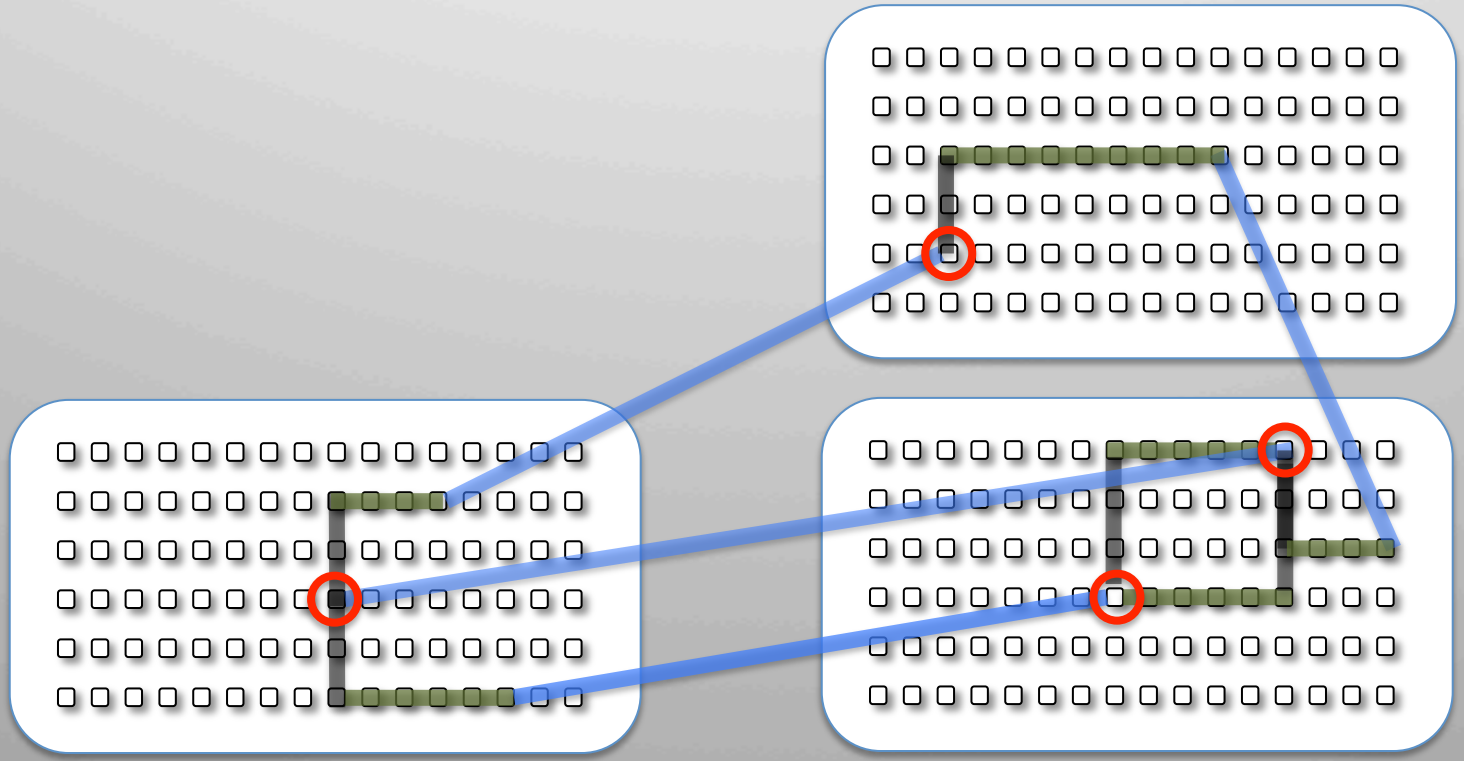
col:
all-to-all



row: all-to-all



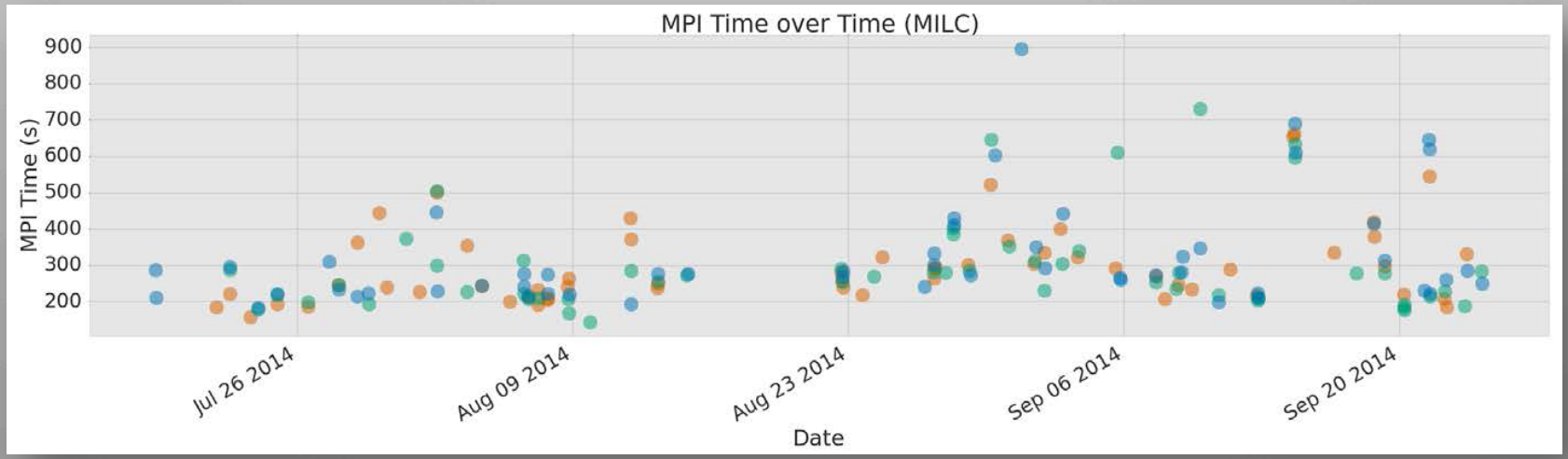
Dragonfly



Unexplained Performance Variations

MIMD Lattice Computation (MILC) for studying quantum chromodynamics (QCD), 4D-stencil communication

Large (>400%) performance variations



Inter-job interference studies

HW Counters

- Can be retrieved only for 'your' routers (*)
- Info only about incoming traffic
- Only aggregated data (e.g. not per job)

Inter-job interference studies on a production machine is problematic

Damselfly

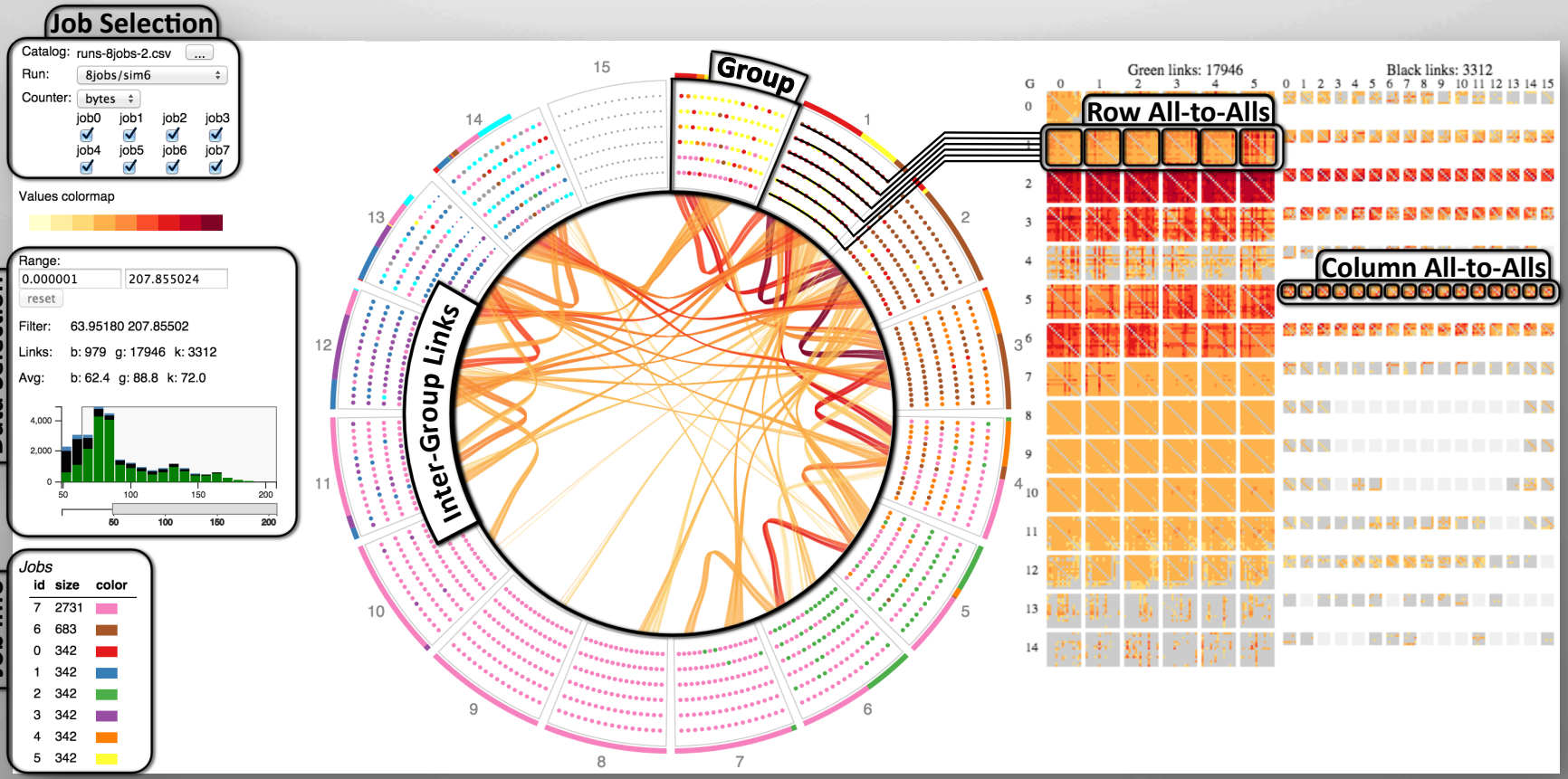
A network simulator that model the steady state behavior of dragonfly networks

Study the effects of

- job placement, jobs size, parallel workloads and network configurations

on network throughput

DnD: Damsels ‘n Dragons



Dragonfly

radial matrix graph

Catalog: runs-edison.csv ...
Run: {512-261}
Counter: flits

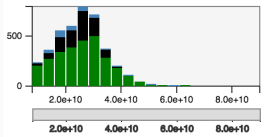
Values colormap

Range:
8e+9 9e+10
reset Freeze

Filter: 8.00000e+9 9.00000e+10

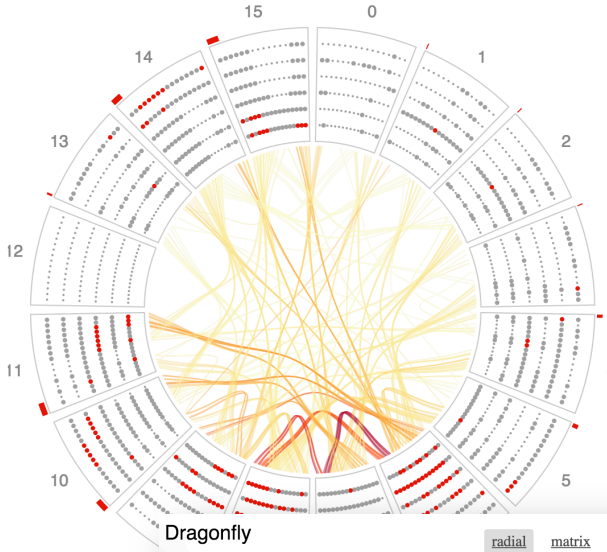
Links: b: 331 g: 2783 k: 945

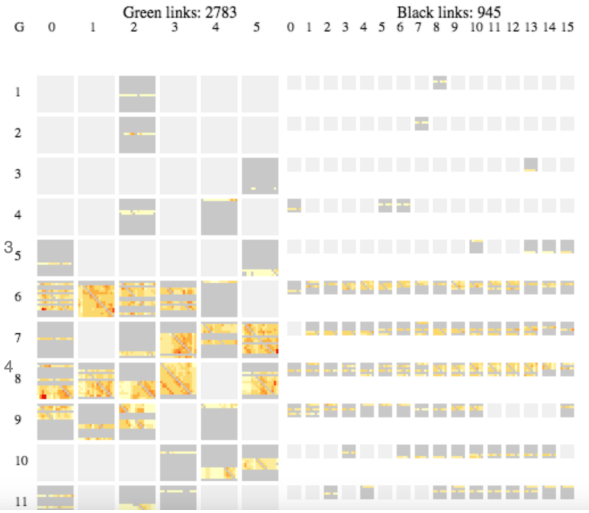
Avg: b: 2.84e+10 g: 2.56e+10 k: 2.50e+10



Jobs

id	size	color
NaN	512	





261 sec

Dragonfly

radial matrix graph

Catalog: runs-edison.csv ...
Run: {512-645}
Counter: flits

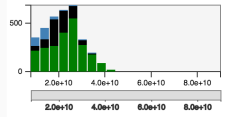
Values colormap

Range:
8e+9 9e+10
reset Freeze

Filter: 8.00000e+9 9.00000e+10

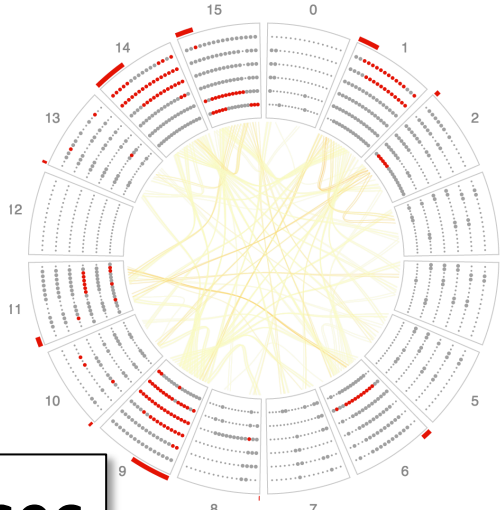
Links: b: 269 g: 2245 k: 805

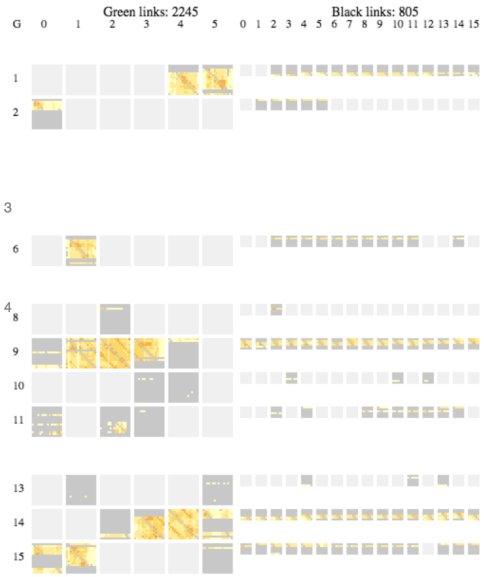
Avg: b: 1.43e+10 g: 2.22e+10 k: 2.07e+10



Jobs

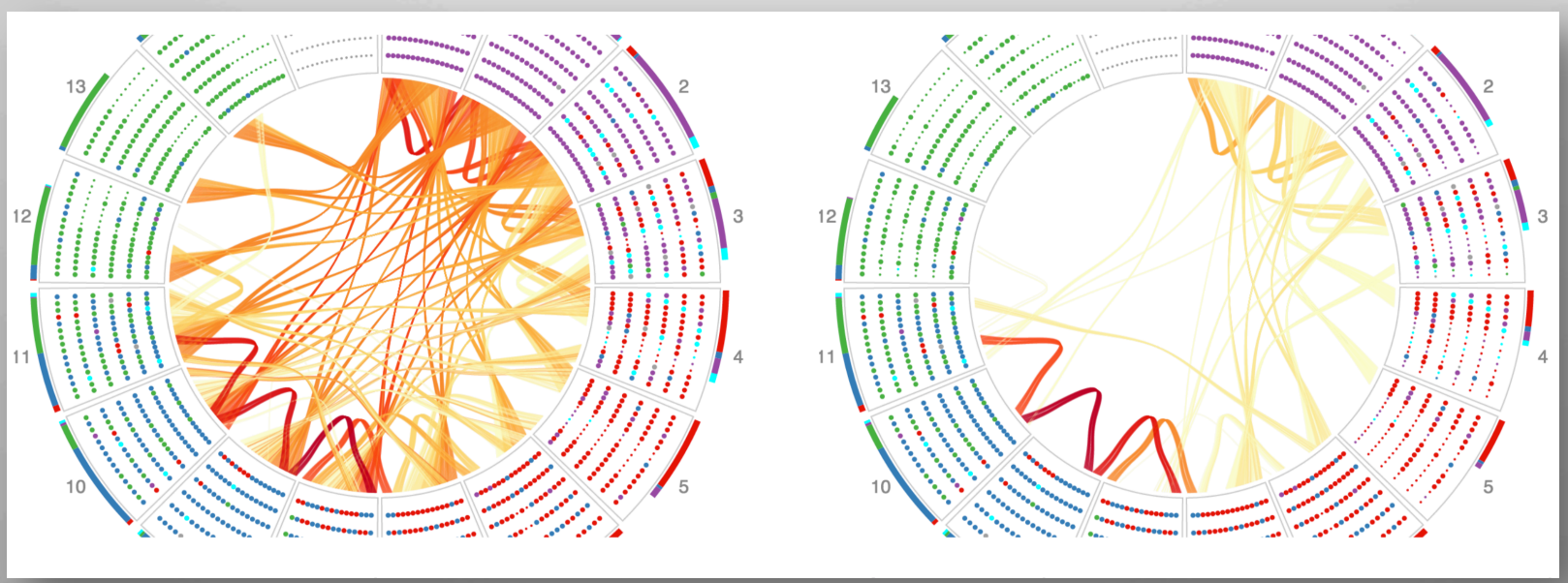
id	size	color
NaN	512	





645 sec

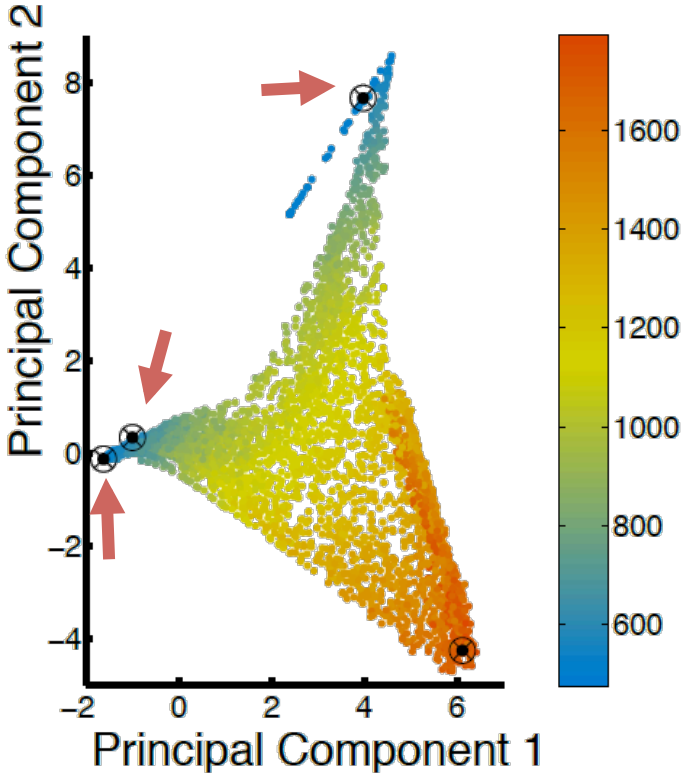
Inter-group links on a dragonfly network for parallel workload

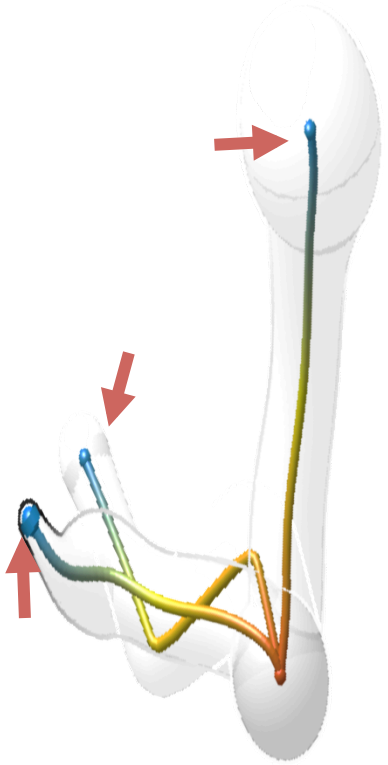


Right: adding inter-group cables reduce network hotspots

Topological Abstractions Techniques for Visualizing and Exploring High Dim. Data

Reduce dimensionality
and then extract structure





Conclusions

We need **better access** on hardware/software info

- at a level that is not currently enabled by the vendors

Simulations will **interfere** more with each other

- compete for scarce resource such as network, I/O and memory hierarchies

Need data at a facility scale

- Simulation level data collection/analysis will be increasingly less meaningful/useful

Hypothesis driven visualization

Currently

- Data collection is completely separated from the analysis and visualization

Long term

- Query and analyze a simulation **while it's running**
  - change data collection on the fly
  - change how it is processed
  - change how it is presented to the user

Final Take Away

Don’t use red green colors

Don’t use a laser pointer

Don’t use pie charts

Do have a dinosaurs in your presentations